Contents

Visualizing Your Mesh

7 minute read page test

This task shows you how to visualize different aspects of your Istio mesh.

As part of this task, you install the Kiali addon and use the web-based graphical user interface to view service graphs of the mesh and your Istio configuration objects.

This task does not cover all of the features provided by Kiali. To learn about the full set of features it

supports, see the Kiali website.

This task uses the Bookinfo sample application as the example throughout. This task assumes the Bookinfo application is installed in the `bookinfo` namespace.

Before you begin

Follow the Kiali installation documentation to deploy Kiali into your cluster.

Generating a graph

1. To verify the service is running in your cluster, run the following command:

   ```
   $ kubectl -n istio-system get svc kiali
   ```

2. To determine the Bookinfo URL, follow the instructions to determine the Bookinfo ingress `GATEWAY_URL`.

3. To send traffic to the mesh, you have three options

   - Visit `http://$GATEWAY_URL/productpage` in your web browser
   - Use the following command multiple times:

```
$ curl http://$GATEWAY_URL/productpage
```

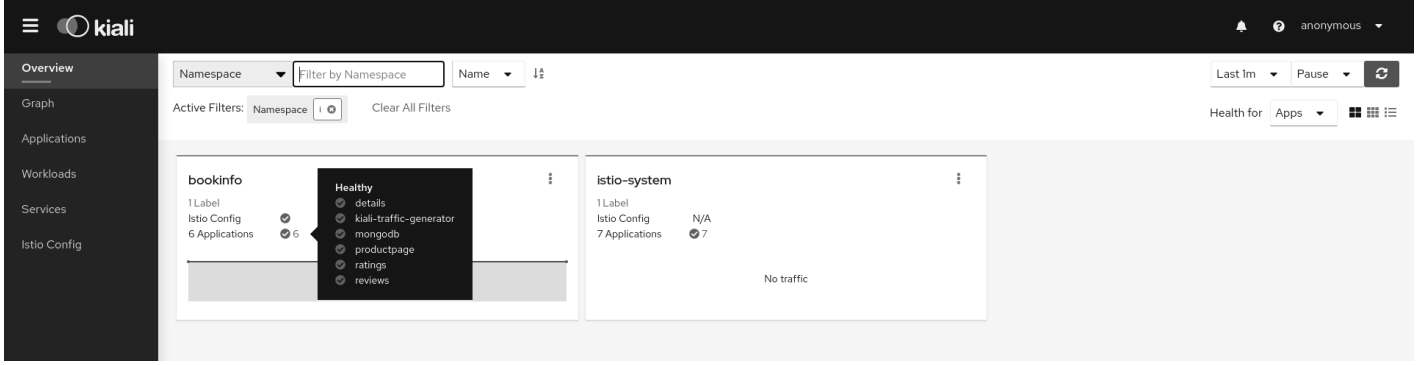
- If you installed the `watch` command in your system, send requests continually with:

```
$ watch -n 1 curl -o /dev/null -s -w %{http_code} $GATEWAY_URL/pro
ductpage
```

4. To open the Kiali UI, execute the following command in your Kubernetes environment:

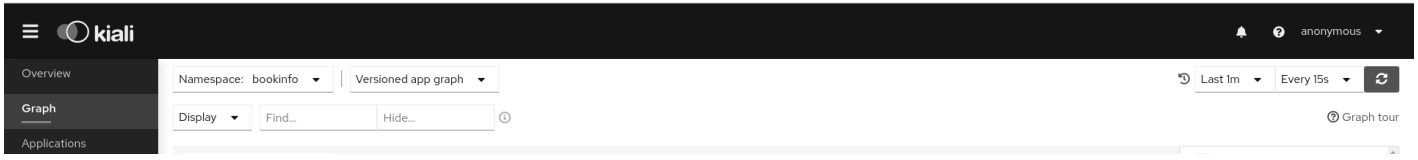
```
$ istioctl dashboard kiali
```

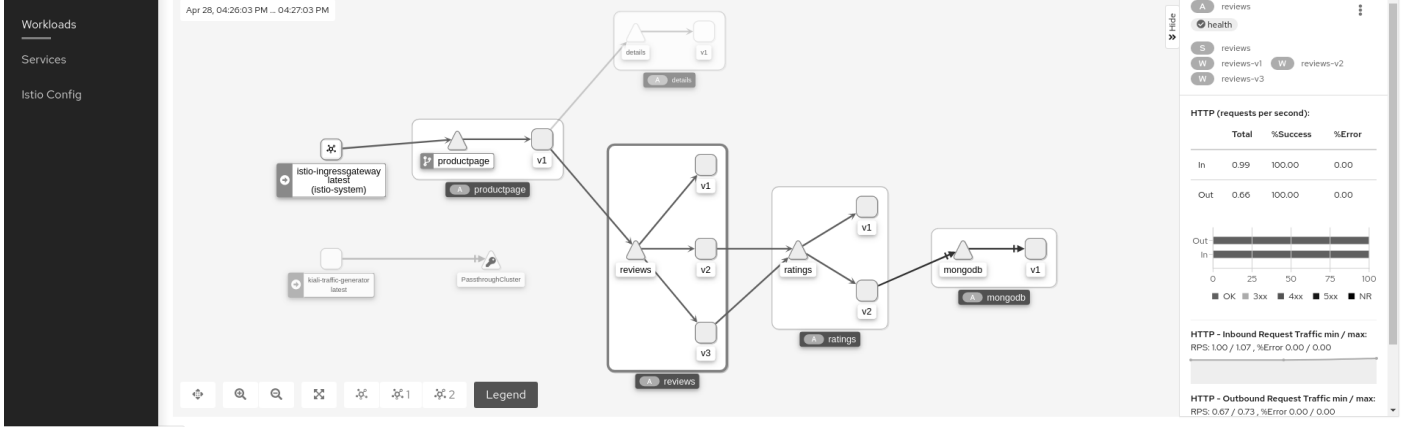
5. View the overview of your mesh in the **Overview** page that appears immediately after you log in. The **Overview** page displays all the namespaces that have services in your mesh. The following screenshot shows a similar page:



Example Overview

6. To view a namespace graph, Select the `Graph` option in the kebab menu of the Bookinfo overview card. The kebab menu is at the top right of card and looks like 3 vertical dots. Click it to see the available options. The page looks similar to:





Example Graph

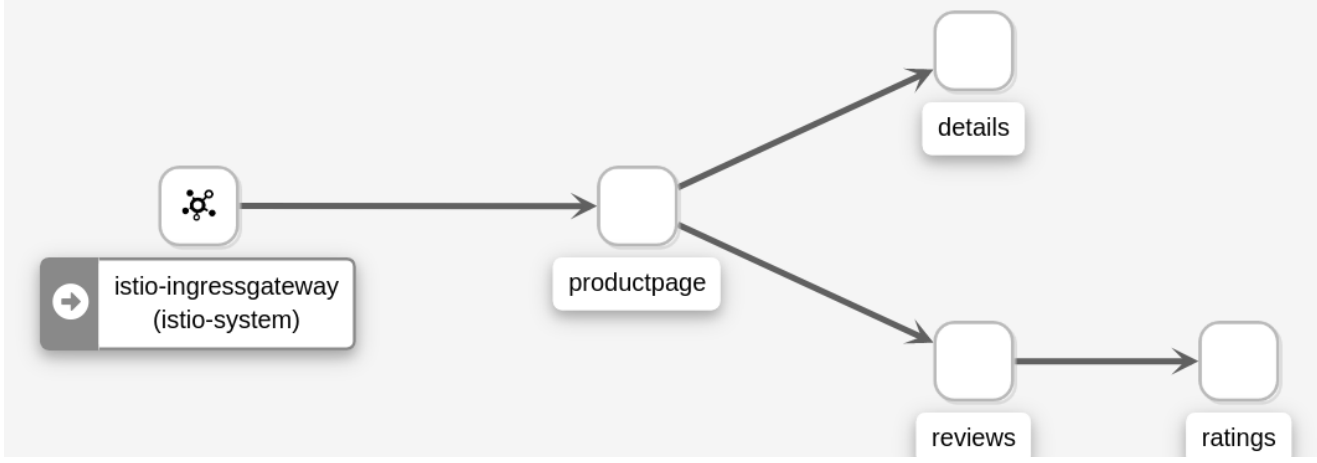
7. The graph represents traffic flowing through the service mesh for a period of time. It is generated using Istio telemetry.

8. To view a summary of metrics, select any node or edge in the graph to display its metric details in the summary details panel on the right.

9. To view your service mesh using different graph types,

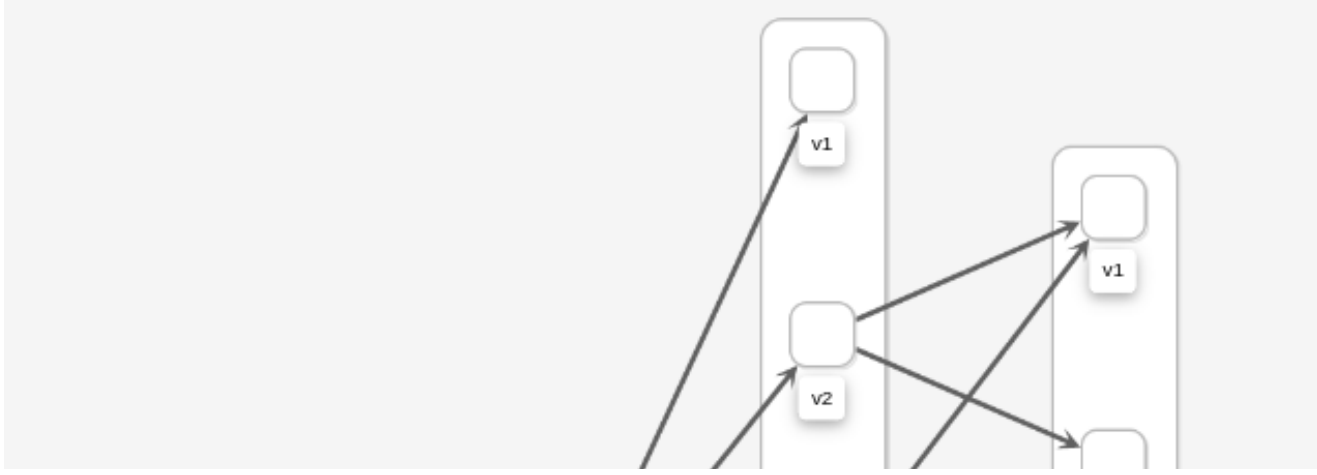
select a graph type from the **Graph Type** drop down menu. There are several graph types to choose from: **App**, **Versioned App**, **Workload**, **Service**.

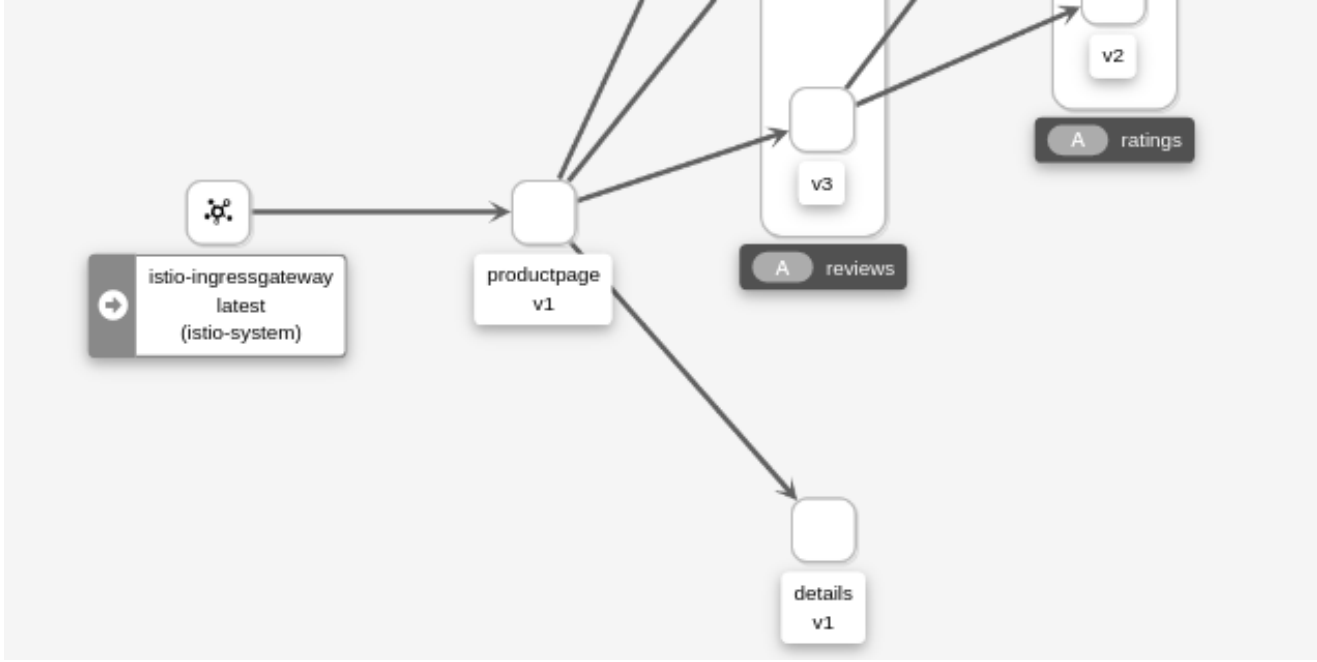
- The **App** graph type aggregates all versions of an app into a single graph node. The following example shows a single **reviews** node representing the three versions of the reviews app. Note that the `Show Service Nodes` Display option has been disabled.



Example App Graph

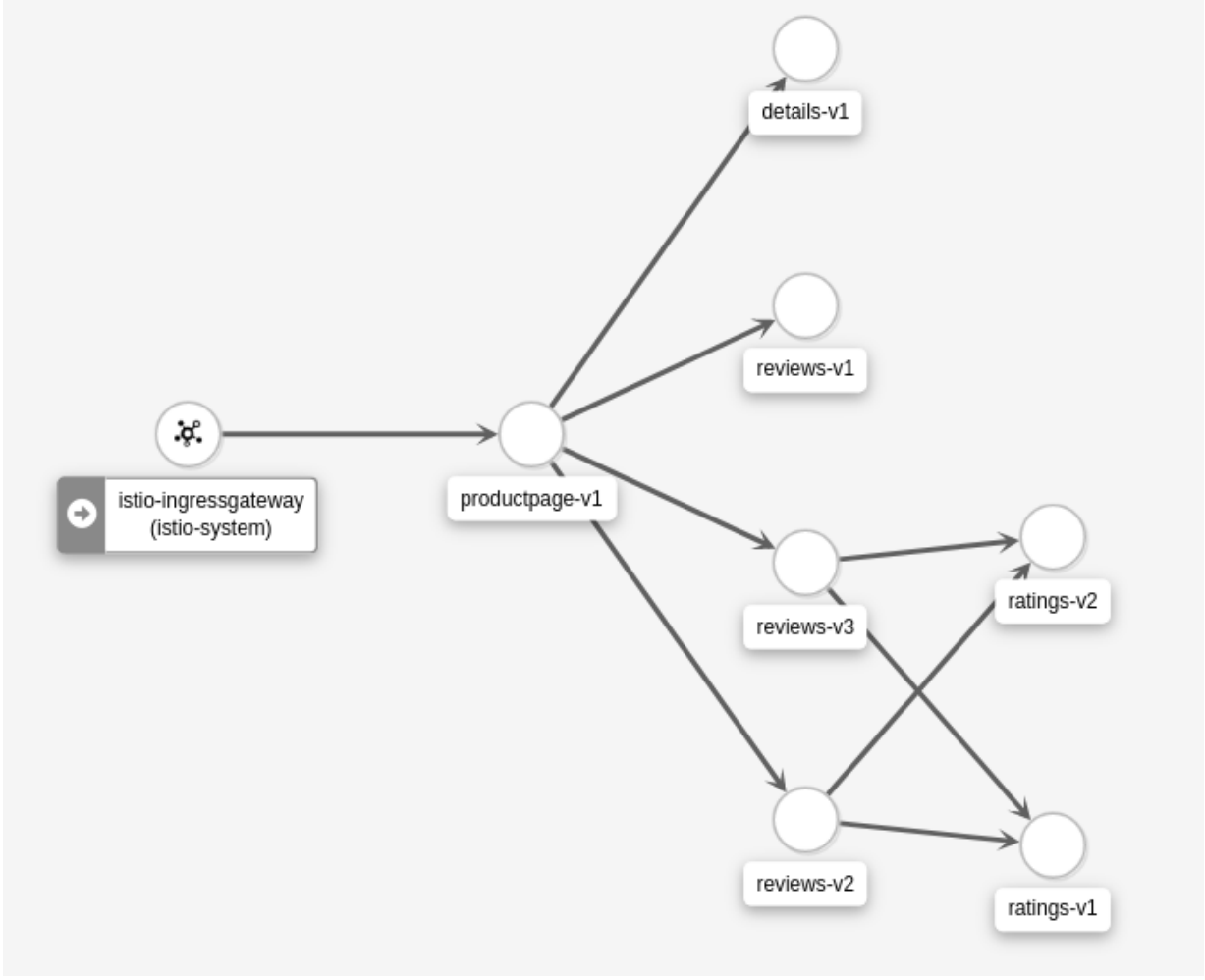
- The **Versioned App** graph type shows a node for each version of an app, but all versions of a particular app are grouped together. The following example shows the **reviews** group box that contains the three nodes that represents the three versions of the reviews app.





Example Versioned App Graph

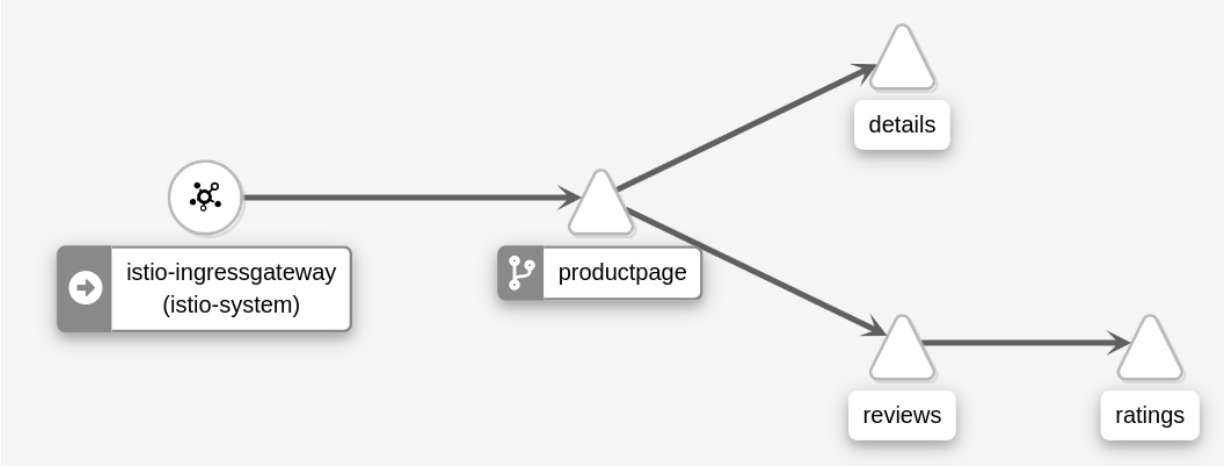
- The **Workload** graph type shows a node for each workload in your service mesh. This graph type does not require you to use the `app` and `version` labels so if you opt to not use those labels on your components, this may be your graph type of choice.



Example Workload Graph

- The **Service** graph type shows a high-level

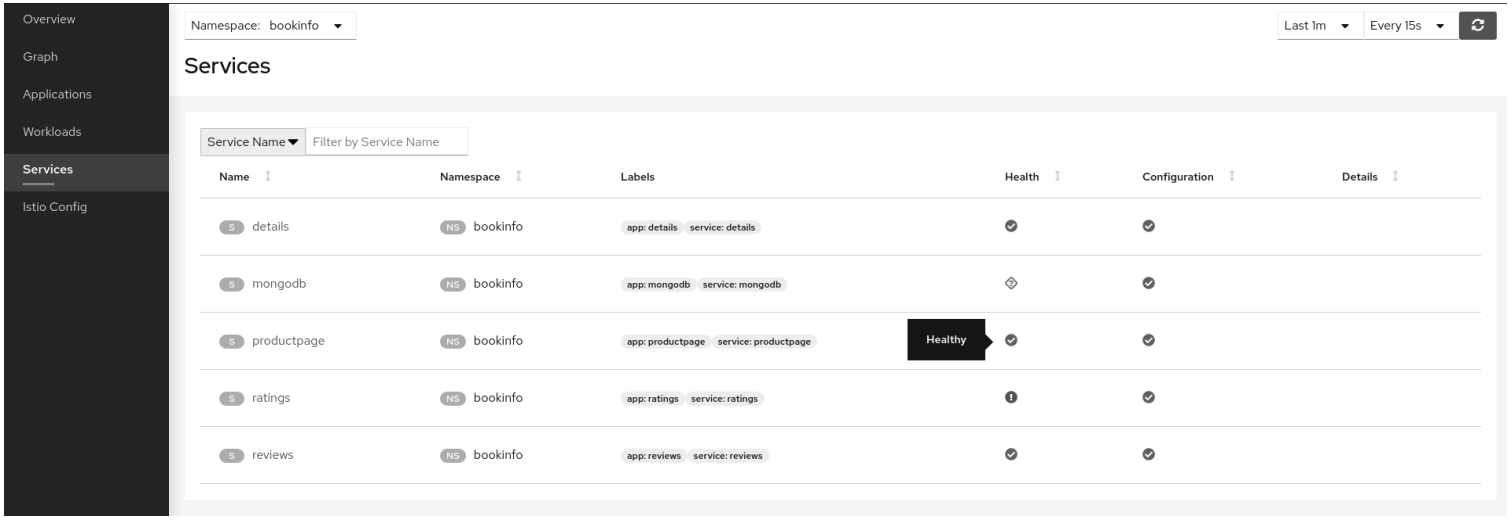
aggregation of service traffic in your mesh.



Example Service Graph

Examining Istio configuration

1. The left menu options lead to list views for **Applications**, **Workloads**, **Services** and **Istio Config**. The following screenshot shows **Services** information for the Bookinfo namespace:

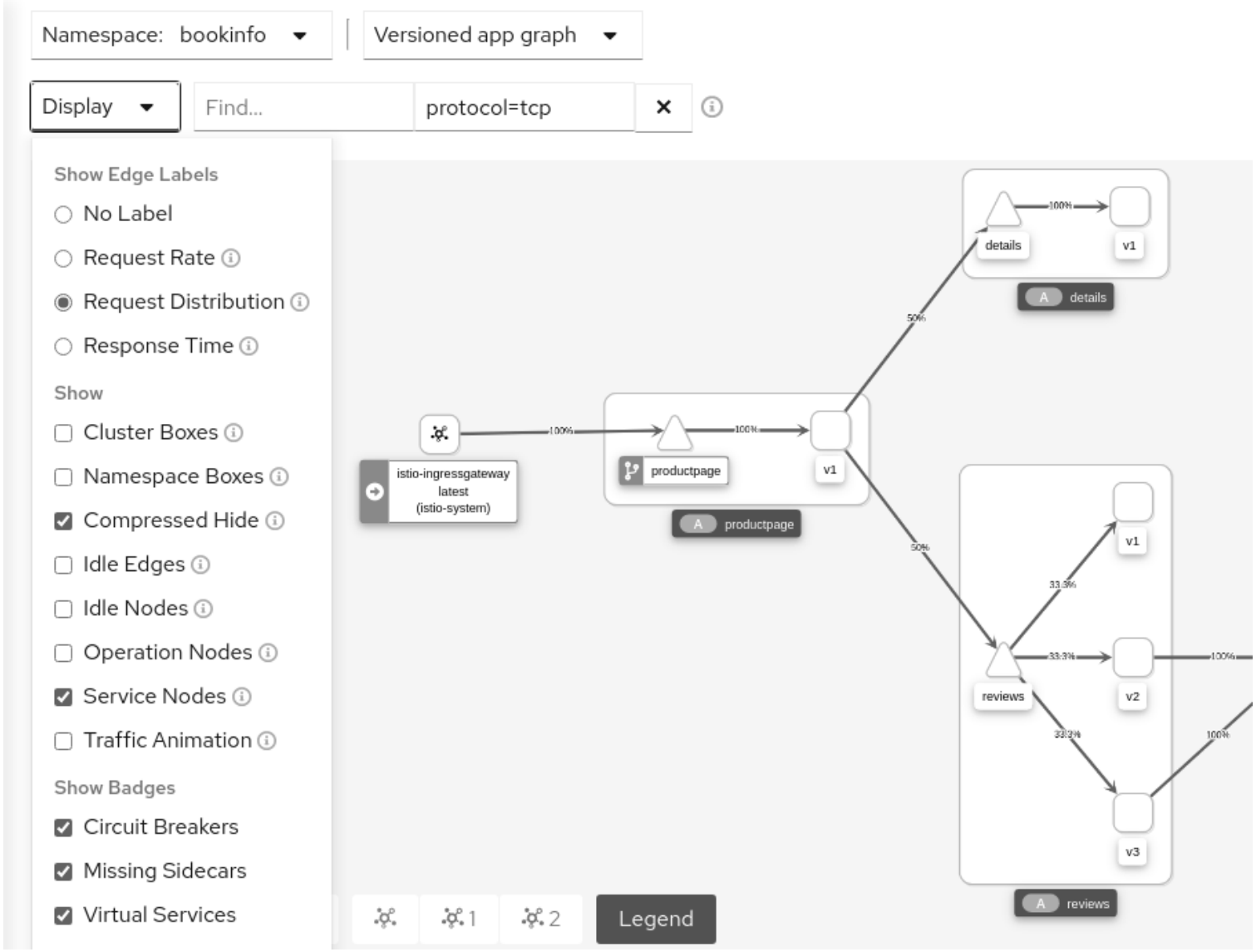


Example Details

Traffic Shifting

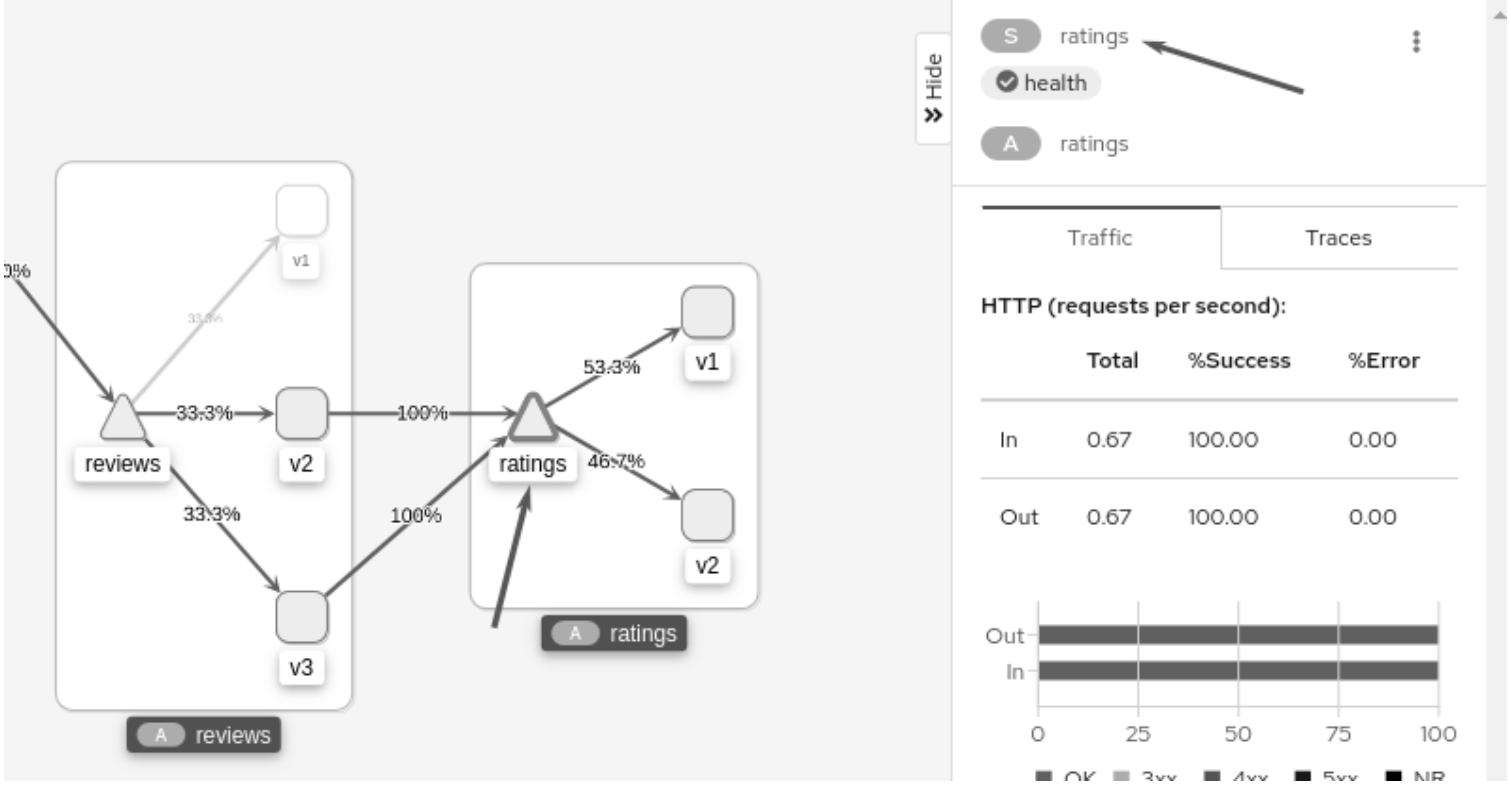
You can use the Kiali traffic shifting wizard to define the specific percentage of request traffic to route to two or more workloads.

1. View the **Versioned app graph** of the `bookinfo` graph.
    - Make sure you have enabled the **Request Distribution** Edge Label **Display** option to see the percentage of traffic routed to each workload.
    - Make sure you have enabled the Show **Service Nodes Display** option to view the service nodes in the graph.



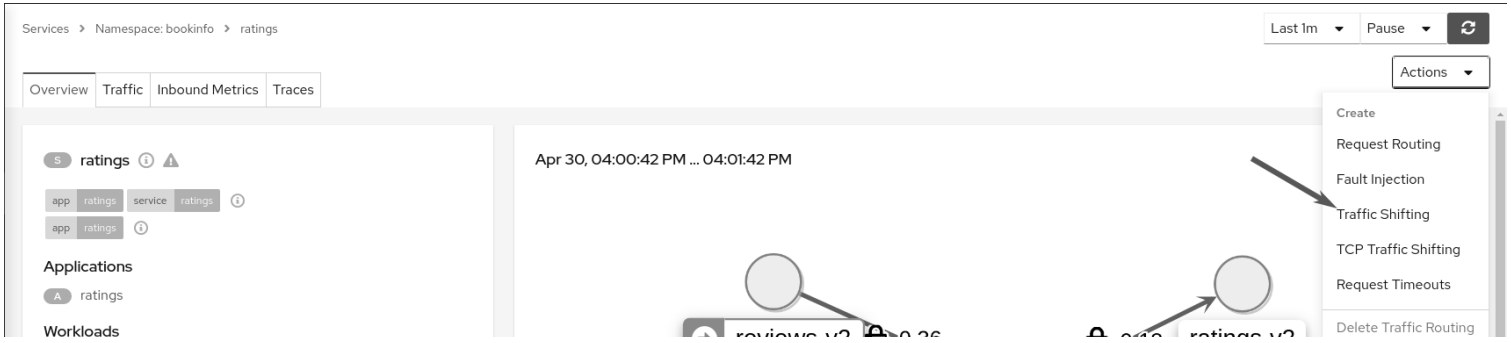
Bookinfo Graph Options

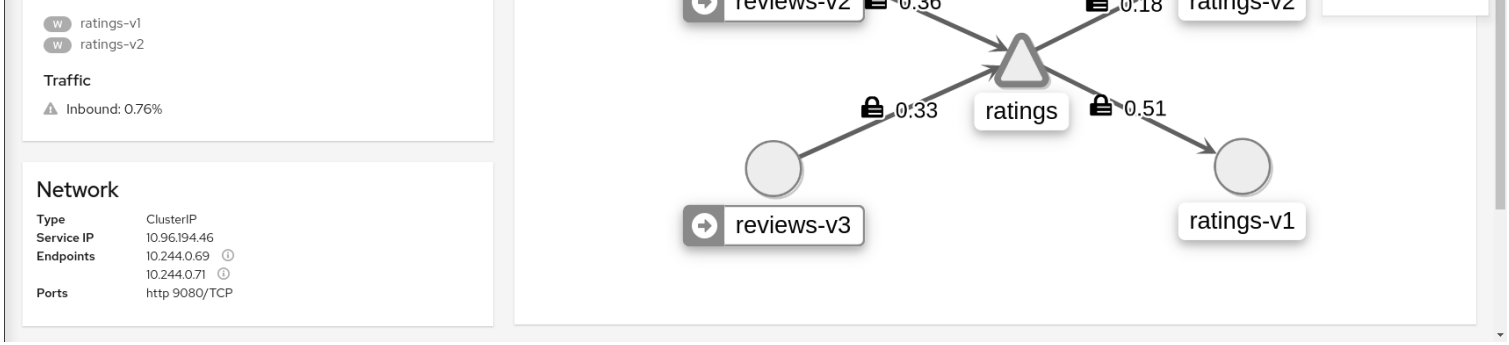
2. Focus on the `ratings` service within the `bookinfo` graph by clicking on the `ratings` service (triangle) node. Notice the `ratings` service traffic is evenly distributed to the two `ratings` workloads `v1` and `v2` (50% of requests are routed to each workload).



Graph Showing Percentage of Traffic

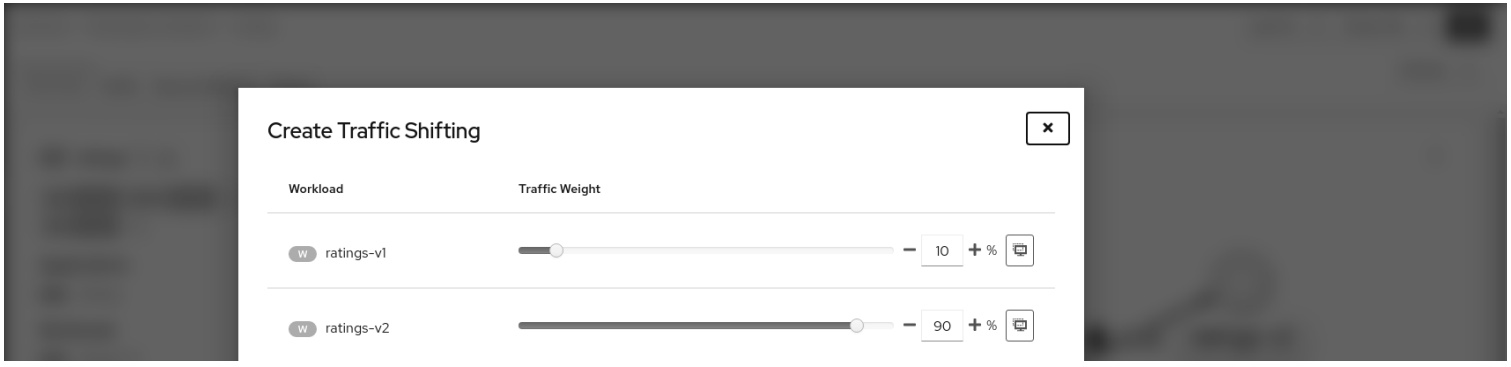
3. Click the **ratings** link found in the side panel to go to the detail view for the `ratings` service. This could also be done by secondary-click on the `ratings` service node, and selecting `Details` from the context menu.
4. From the **Actions** drop down menu, select **Traffic Shifting** to access the traffic shifting wizard.

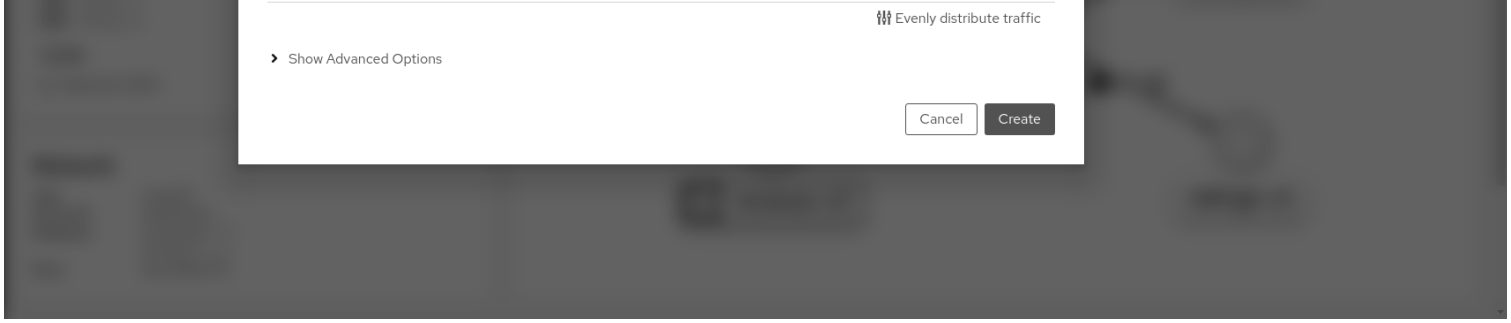




Service Actions Menu

5. Drag the sliders to specify the percentage of traffic to route to each workload. For `ratings-v1`, set it to 10%; for `ratings-v2` set it to 90%.



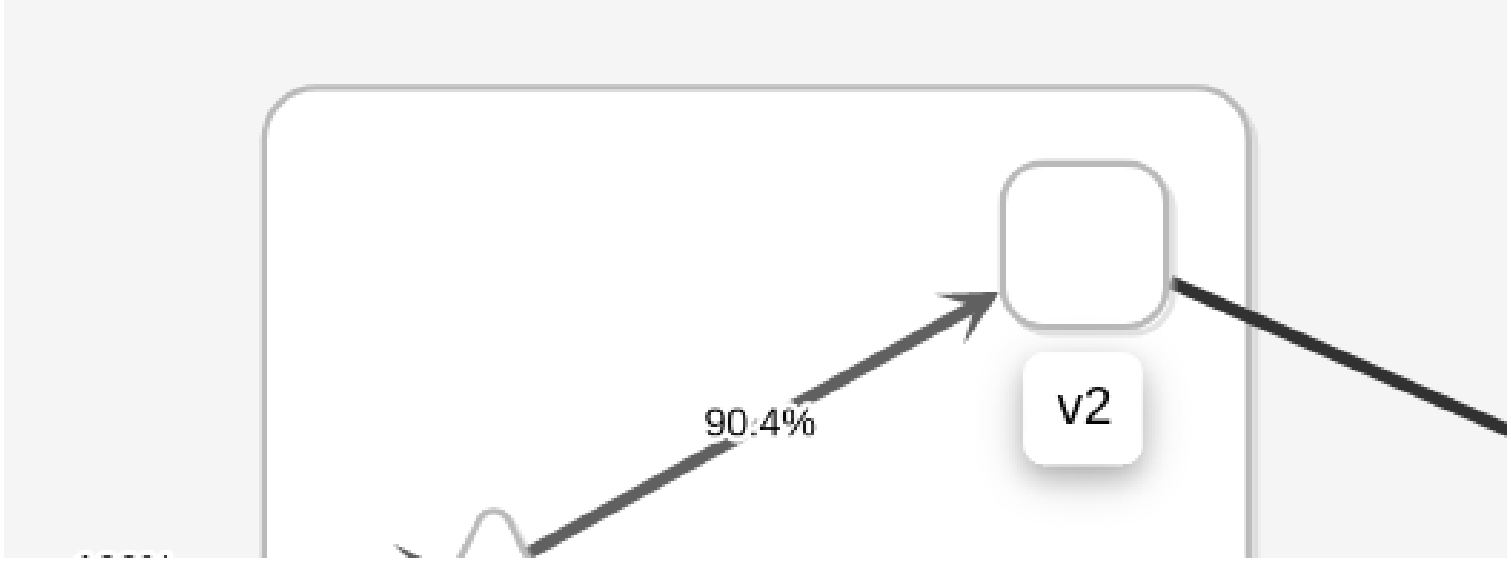


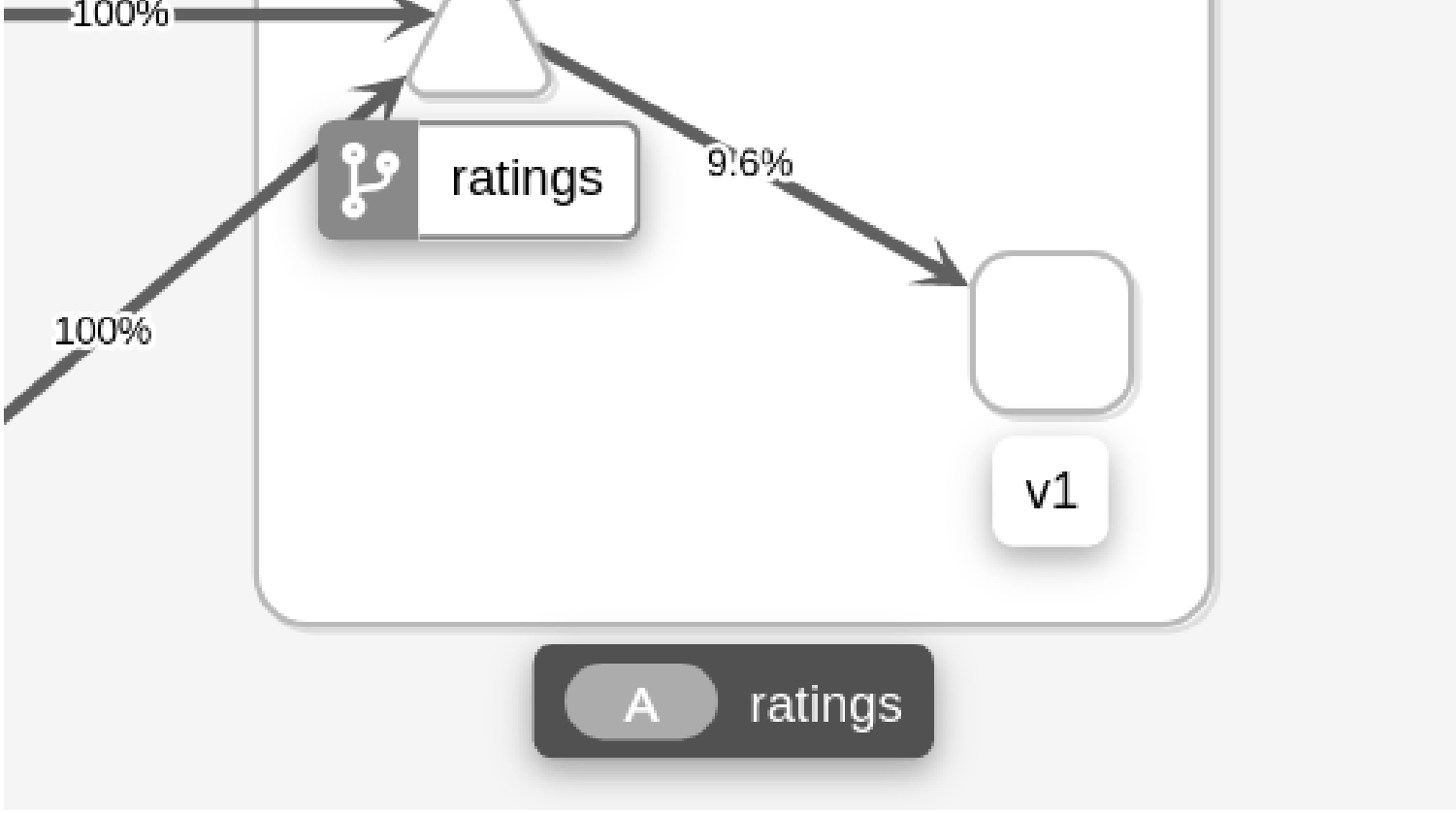
Weighted Routing Wizard

6. Click the **Create** button to apply the new traffic settings.
7. Click **Graph** in the left hand navigation bar to return to the `bookinfo` graph. Notice that the `ratings` service node is now badged with the `virtual service` icon.
8. Send requests to the `bookinfo` application. For example, to send one request per second, you can execute this command if you have `watch` installed on your system:

```
$ watch -n 1 curl -o /dev/null -s -w %{http_code} $GATEWAY_URL/productpage
```

9. After a few minutes you will notice that the traffic percentage will reflect the new traffic route, thus confirming the fact that your new traffic route is successfully routing 90% of all traffic requests to `ratings-v2`.





90% Ratings Traffic Routed to ratings-v2

Validating Istio configuration

Kiali can validate your Istio resources to ensure they follow proper conventions and semantics. Any problems detected in the configuration of your Istio resources can be flagged as errors or warnings depending on the severity of the incorrect configuration. See the Kiali validations page for the list of all validation checks Kiali performs.

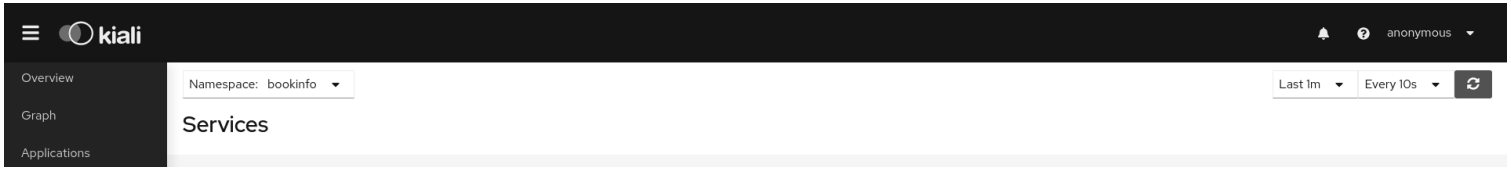
Istio provides `istioctl analyze` which provides analysis in a way that can be used in a CI pipeline. The two approaches can be complementary.

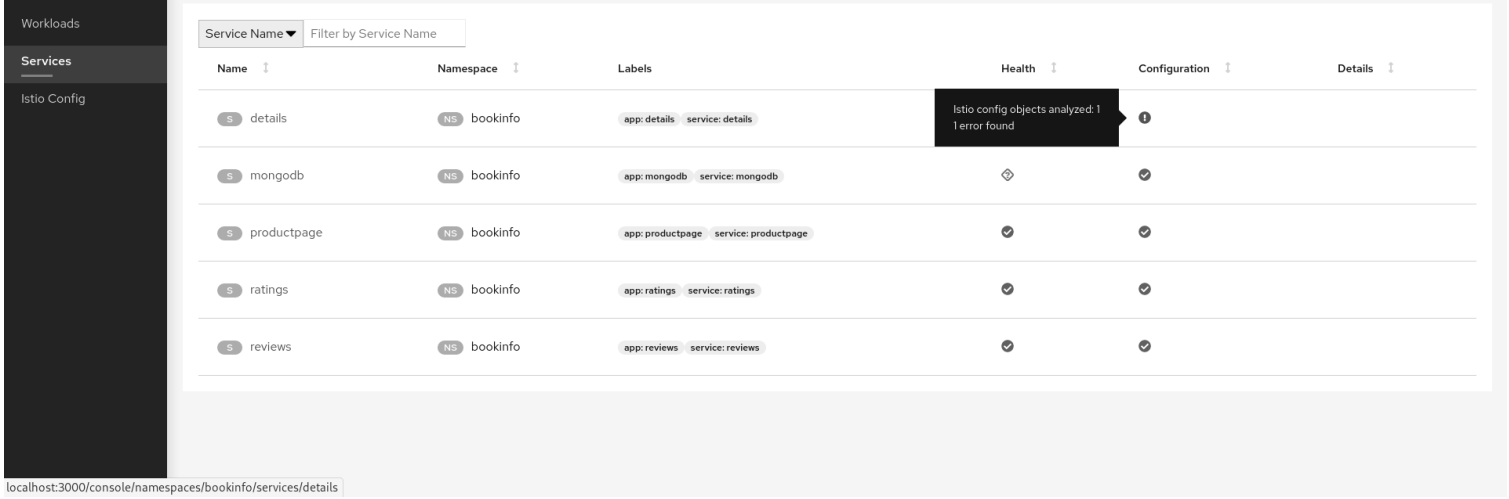
Force an invalid configuration of a service port name to see how Kiali reports a validation error.

1. Change the port name of the `details` service from `http` to `foo`:

```
$ kubectl patch service details -n bookinfo --type json -p '[{"op":"re
place","path":"/spec/ports/0/name", "value":"foo"}]'
```

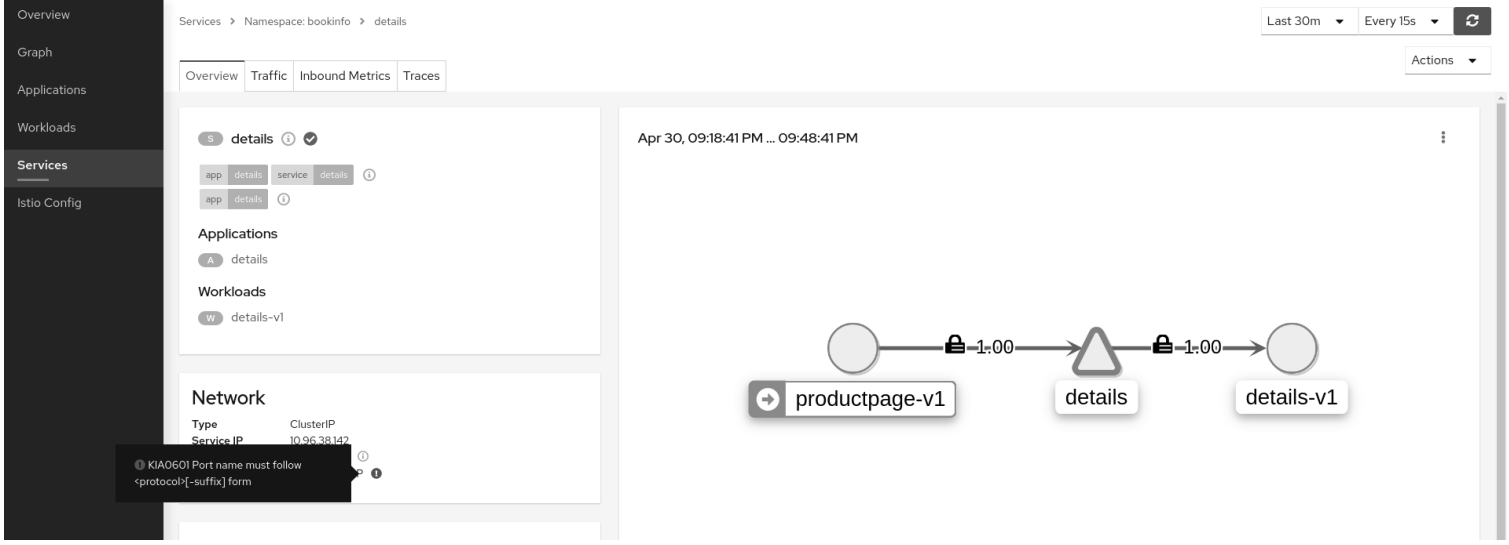
2. Navigate to the **Services** list by clicking **Services** on the left hand navigation bar.
3. Select `bookinfo` from the **Namespace** drop down menu if it is not already selected.
4. Notice the error icon displayed in the **Configuration** column of the `details` row.





Services List Showing Invalid Configuration

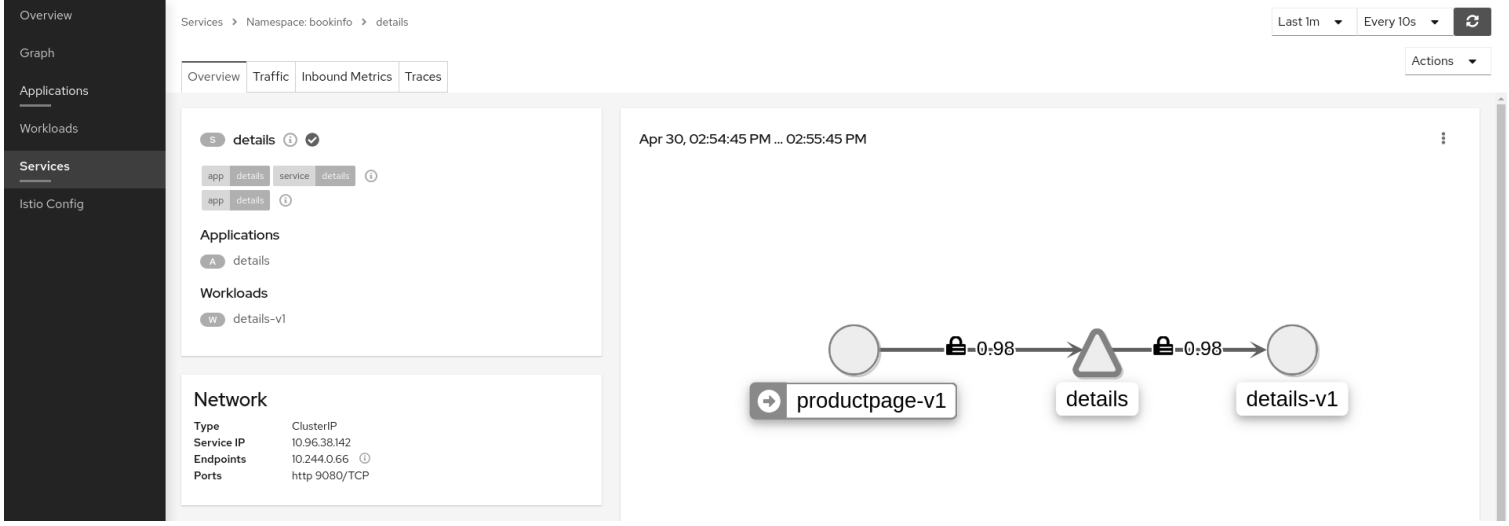
5. Click the **details** link in the **Name** column to navigate to the service details view.
6. Hover over the error icon to display a tool tip describing the error.



Service Details Describing the Invalid Configuration

7. Change the port name back to `http` to correct the configuration and return `bookinfo` back to its normal state.

```
$ kubectl patch service details -n bookinfo --type json -p '[{"op":"re
place","path":"/spec/ports/0/name", "value":"http"}]'
```



Service Details Showing Valid Configuration

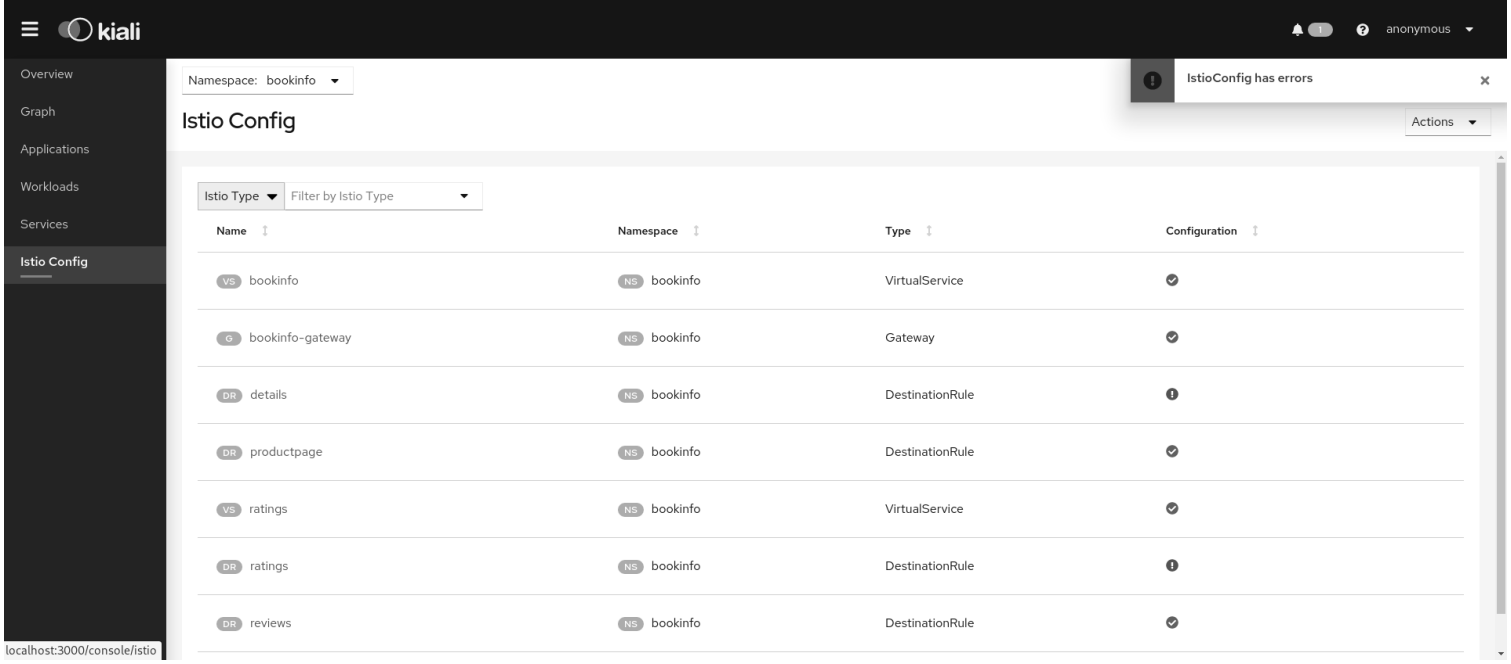
Viewing and editing Istio configuration YAML

Kiali provides a YAML editor for viewing and editing Istio configuration resources. The YAML editor will also provide validation messages when it detects incorrect configurations.

1. Create Bookinfo destination rules:

```
$ kubectl -n bookinfo apply -f @samples/bookinfo/networking/destinatio
n-rule-all.yaml@
```

2. Click `Istio Config` on the left hand navigation bar to navigate to the Istio configuration list.
3. Select `bookinfo` from the **Namespace** drop down menu if it is not already selected.
4. Notice the error message and the error icons that alert you to several configuration problems.



Istio Config List Incorrect Configuration Messages

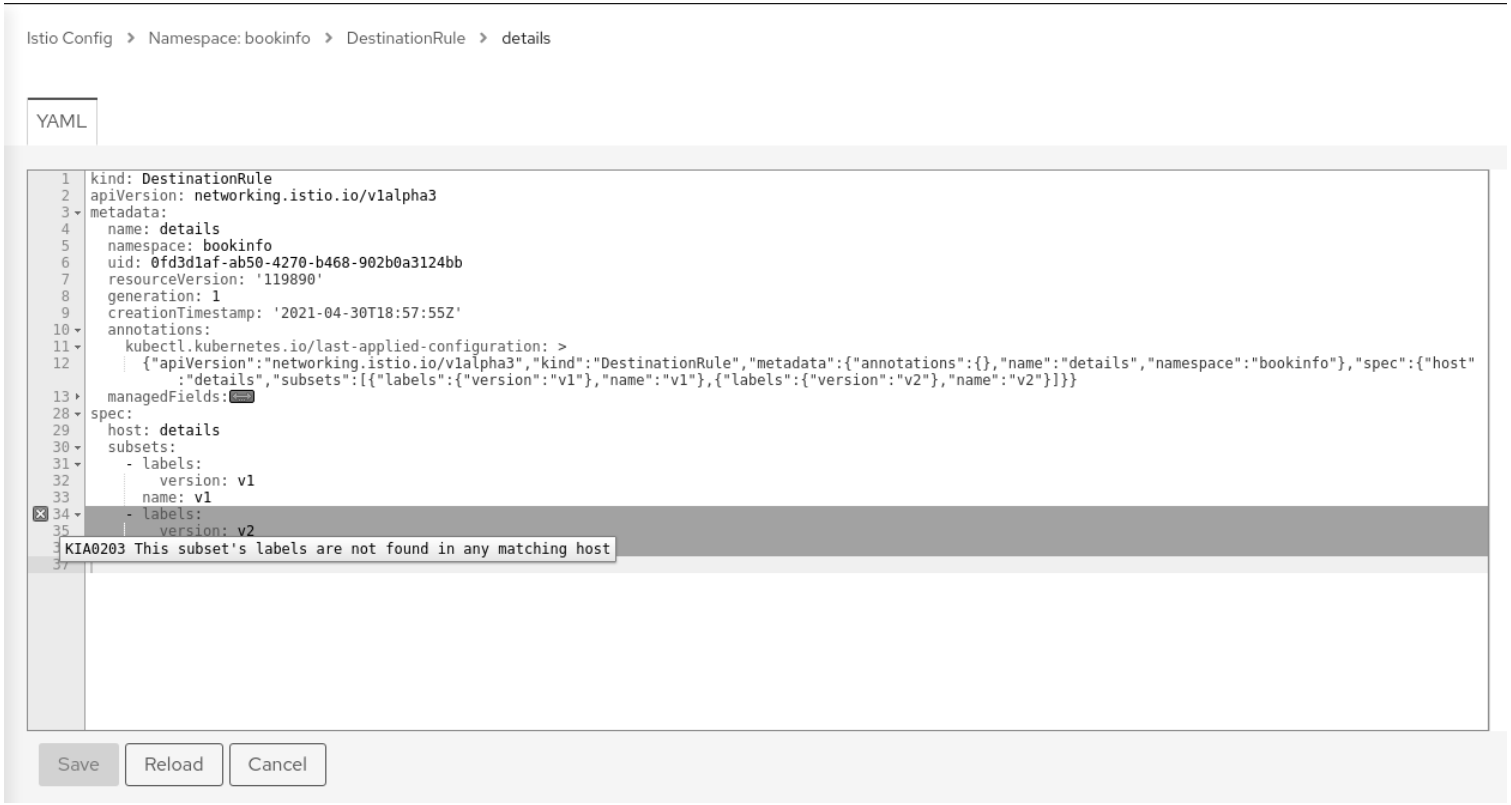
5. Click the error icon in the **Configuration** column of the `details` row to navigate to the `details` destination rule view.
6. The **YAML** tab is preselected. Notice the color highlights

and icons on the rows that have failed validation checks.

YAML Editor Showing Validation Errors and Warnings

7. Hover over the red icon to view the tool tip message that

informs you of the validation check that triggered the error. For more details on the cause of the error and how to resolve it, look up the validation error message on the Kiali Validations page.



YAML Editor Showing Error Tool Tip

8. Delete the destination rules to return `bookinfo` back to its original state.

```
$ kubectl -n bookinfo delete -f samples/bookinfo/networking/destinatio
n-rule-all.yaml
```

Additional Features

Kiali has many more features than reviewed in this task, such as an integration with Jaeger tracing.

For more details on these additional features, see the Kiali documentation.

For a deeper exploration of Kiali it is recommended to run through the Kiali Tutorial.

Cleanup

If you are not planning any follow-up tasks, remove the Bookinfo sample application and Kiali from your cluster.

1. To remove the Bookinfo application, refer to the Bookinfo

cleanup instructions.

2. To remove Kiali from a Kubernetes environment:

```
$ kubectl delete -f https://raw.githubusercontent.com/istio/istio/release-1
.11/samples/addons/kiali.yaml
```